Amendments to the Claims:

1 through 36 (previously canceled)

37. (previously presented) A method for generating graphics by aliasing vertex attributes during vertex processing, the method comprising:

associating an identifier defined by an application-programmable vertex processing system with a parameter defined according to conventional vertex processing;

calling the identifier defined by the application-programmable vertex processing system;

determining which parameter defined according to conventional vertex processing is associated with the called identifier; and

retrieving the parameter determined to be associated with the called identifier.

38. (previously presented) The method of claim 37, wherein the parameter defined according to conventional vertex processing comprises an OpenGL defined parameter.

39. (previously presented) The method of claim 37, wherein the parameter defined according to conventional vertex processing comprises a D3D defined parameter.

40. (previously presented) The method of claim 37, further comprising:

retrieving a second parameter defined according to conventional vertex processing by calling the convention-defined vertex parameter.

41. (previously presented) The method of claim 37, wherein the identifier defined by an application-programmable vertex processing system corresponds to a vertex attribute register.

42. (previously presented) A system for generating graphics, the system comprising:

a convention-defined vertex processing system;

an application-programmable vertex processing system; and

a memory device configured to store an identifier defined by the application-programmable vertex processing system with a parameter defined within the convention-defined vertex processing system.

43. (previously presented) A system for generating graphics by aliasing vertex attributes during vertex processing, the system comprising:

a memory device;

a plurality of instructions stored on the memory device, the plurality of instructions configured to:

associate an identifier defined by an application-programmable vertex processing system with a parameter defined according to conventional vertex processing;

call the identifier defined by the application-programmable vertex processing system;

determine which parameter defined according to conventional vertex processing is associated with the called identifier; and

retrieve the parameter determined to be associated with the called identifier.

44. (previously presented) A system for generating graphics by aliasing vertex attributes during vertex processing, the system comprising:

means for associating an identifier defined by an application-programmable vertex processing system with a parameter defined according to conventional vertex processing;

means for calling the identifier defined by the application-programmable vertex processing system;

means for determining which parameter defined according to conventional vertex processing is associated with the called identifier; and

means for retrieving the parameter determined to be associated with the called identifier.

45. (new) A method for generating graphics by aliasing vertex attributes during vertex processing, the method comprising:

associating an identifier defined by an application-programmable vertex processing system with a parameter defined according to a conventional vertex processing system;

calling the identifier defined by the application-programmable vertex processing system;

looking up the parameter defined according to the conventional vertex processing system that matches the called identifier;

retrieving the parameter determined to be associated with the called identifier; and

rendering a graphic using the retrieved parameter.